Exemplo:

```
program [illegible] = (
    VERDADEIRO =
    SIM = true;
    NULL(FALSO) = false;
begin  READ (ch);
       [illegible] integer;
       ACHOU, [illegible];
       ERASE, LETRA = char;
```

Cada programa só terá uma única palavra-chave [illegible] e ele. O comentário deverá sair [illegible] Comentário pode estar no meio [illegible]

Como o PASCAL não possui o operador de exponenciação, vamos operacionalizá-lo através do programa PASCAL capaz de ler dois números x e y e imprimir o valor de x^{y}:

```
program exponenciação (input, output);
  {exponenciação com expoente inteiro};
  var x, y, u, z: real;
  begin
    read (x, y);
    writeln (x, y);
    z := 1;
    u := x;
    e := y;
    while e > 0 do
    begin                          {z*u^e*e = x^y}
      while not odd (e) do
      begin
        e := ediv2;
        u := sqr (u)
      end;
      e := e - 1;
      z := u*z
    end;
    writeln (z)        {z = x^y}
end.
```

```
1.0000000000e+00   7   1.280000000000e+02   <- valores
        y                  x                   impressos (exemplo)
```


c) VOLUMES DADOS HOMOGÊNEOS

```
PASCAL
  type x = array [1..5] of integer;
  var FATORIAL
      M = array [1..3, 1..3] of real
  var VENDAS

PASCAL
  type V = array [1..5] of integer;
      M = array [V..V, V..V] of real;
  var FATORIAL: V;
      VENDAS: M;
```

Temos, portanto, quatro maneiras de declarar PEÇOAS como array. Observe que a [illegible] da variável dos limites em cada dimensão é feita por dois pontos (ex: [illegible]) [illegible] como no PORTUGOL [illegible].

O exemplo abaixo é um programa PASCAL que trata com arrays.

```
program ANAVEN;
  {relatório de vendas de empresa entre 1978 e 1980}
  type VE = array [1978..1980] of integer;
  var VENDAS, LUCROS: VE
      I: integer;
      MÉDIA: real;
  begin
    for I := 1978 to 1980 do
      read (VENDAS [I]);
    MÉDIA := (VENDAS [1978] + VENDAS [1979] + VENDAS [1980]) / 3;
    for I := 1978 to 1980 do
      LUCROS [I] := MÉDIA - VENDAS [I] * 10;
    for I := 1978 to 1980 do
      begin
        writeln (LUCROS [I], VENDAS [I])
        writeln
      end;
    writeln
  end.
```

d) PROCEDIMENTOS

```
PORTUGOL
  procedimento FATORIAL (N, FAT)
    inteiro: N, FAT;
    início
      | inteiro: I;
      | FAT := 1;
      | para I de 1 até N faça
      |   | FAT := FAT * I;
      | fim para
    fim  {FATORIAL}
```